REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	November 1990	Final, from August to December 1989

4. TITLE AND SUBTITLE
Image Recovery by Simulated Annealing

5. FUNDING NUMBERS
PE: 61102.H44

6. AUTHOR(S)
James B. Cole and David H. Gerstman

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Harry Diamond Laboratories
2800 Powder Mill Road
Adelphi, MD 20783-1197

8. PERFORMING ORGANIZATION REPORT NUMBER
HDL-TM-90-21

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
U.S. Army Laboratory Command
2800 Powder Mill Road
Adelphi, MD 20783-1145

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES
AMS code: 611102.AH440011
HDL PR: AE1052

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

We studied the restoration of noisy binary images of different shapes by simulated annealing. Using the techniques of Bayesian analysis and the Markov random field model for images, the restoration problem was cast into the form of a combinatorial optimization problem that was solved by simulated annealing. Our algorithm was able to halve noise levels of up to 30 percent, and the effectiveness of the algorithm was independent of shape.

14. SUBJECT TERMS
Simulated annealing, Boltzmann distribution, image restoration, image recovery, combinatorial optimization

15. NUMBER OF PAGES
11

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	17. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL

NSN 7540-01-280-5500

Standard Form 298 (Rev 2-89)
Prescribed by ANSI Std. Z39-18
298-102

Contents

Figures

Tables

1. Introduction

The basic problem of image recovery and pattern recognition is to determine the original pattern, **f**, given its corrupted version, **g**. The unknown pattern **f** is an element of the set $S = \{\mathbf{f}_1, \mathbf{f}_2, \mathbf{f}_3, ..\}$, and the task is to deduce which pattern in S gave rise to the image data, **g**. S is called the solution candidate space and could be, for example, the set of alphabetical symbols.

If it is known that certain elements of S have a higher probability of occurring than others (such as alphabetical symbols in text), this *a priori* information can be incorporated into the procedure for finding **f** according to the techniques of Bayesian analysis [1].

In the general image recovery problem, S is the set of all possible patterns on an $n \times n$ pixel image, and the relationship between the original image, **f**, and the image data, **g**, can be modeled by

$$\mathbf{g} = \mathbf{f} + \mathbf{w}, \qquad (1)$$

where **w** is random noise. Sets of pixel brightnesses at lattice position (i, j) are described by $\mathbf{f} = \{f_{ij}\}$, $\mathbf{g} = \{g_{ij}\}$, and $\mathbf{w} = \{w_{ij}\}$. Since **w** is a random variable, **f** cannot be determined from **g** using equation (1). The best we can do is find the pattern $\mathbf{f}^* \in S$ such that $p(\mathbf{f} \mid \mathbf{g})$ is a maximum, where $p(\mathbf{f} \mid \mathbf{g})$ is the probability that, given the image data **g**, **f** was the original image. We cannot, even in principle, be sure that $\mathbf{f}^* = \mathbf{f}$ or is even "close" to it. If, however, **f** is known "in advance" (using known test images that have been corrupted by noise), the "goodness" of $\mathbf{f}^*$ can be evaluated. This is what we have done.

Image recovery and pattern recognition problems are thus combinatorial optimization problems, in which a solution candidate space, S, must be searched. The larger S is, the more difficult the search. The number, N, of images possible on an $n \times n$ pixel array where each pixel has L possible grey levels is given by

$$N = L^{n^2} . \qquad (2)$$

Even for a simple 10×10 binary image where $L = 2$ and $n = 10$, $N = 2^{100}$, so calculating $p(\mathbf{f} \mid \mathbf{g})$ for each possible **f** is computationally infeasible, even in simple cases. A problem whose computational difficulty grows exponentially with some characteristic problem size measure is called NP (nonpolynomial) complete [2]. Most NP complete problems can only be solved by selectively sampling elements of the solution candidate space.

2. Bayesian Analysis

The probability of "event" **f**, given that **g** has occurred, $p(\mathbf{f} \mid \mathbf{g})$, is called a conditional probability. The probability of **f** alone, $p(\mathbf{f})$, and that of **g** alone, $p(\mathbf{g})$, may overlap in "probability space" as illustrated in figure 1. Obviously, $p(\mathbf{f} \mid \mathbf{g})$ and $p(\mathbf{g} \mid \mathbf{f})$ are given by

$$p(\mathbf{f} \mid \mathbf{g}) = \frac{p(\mathbf{f} \wedge \mathbf{g})}{p(\mathbf{g})} \qquad p(\mathbf{g} \mid \mathbf{f}) = \frac{p(\mathbf{f} \wedge \mathbf{g})}{p(\mathbf{f})} , \qquad (3)$$

where $p(\mathbf{f} \wedge \mathbf{g})$ is the probability of both **f** and **g**. Eliminating $p(\mathbf{f} \wedge \mathbf{g})$ from (3) yields

$$p(\mathbf{f} \mid \mathbf{g}) = \frac{p(\mathbf{g} \mid \mathbf{f})\, p(\mathbf{f})}{p(\mathbf{g})} . \qquad (4)$$

Equation (4) expresses Bayes' theorem.

The probability $p(\mathbf{f})$ expresses our *a priori* (independent of the image data, **g**) knowledge of **f**, whereas $p(\mathbf{f} \mid \mathbf{g})$ represents our *a posteriori* (given the image data) information about **f**. Thus $\mathbf{f}^* \in S$ which maximizes $p(\mathbf{f} \mid \mathbf{g})$ is the best fit to both the *a priori* and the *a posteriori* knowledge about **f**. Bayes' theorem allows us to incor-

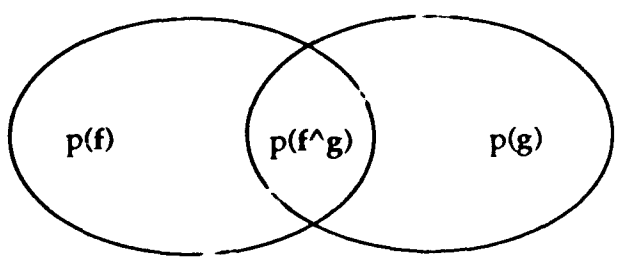

Figure 1. Probability space for Bayes' Theorem.

porate the "experimental data," **g**, with our prior knowledge of **f**. This approach to the image restoration and pattern recognition problem is often called the maximum *a posteriori* (MAP) technique.

3. Combinatorial Optimization Problem

We are now ready to obtain an analytic expression for $p(\mathbf{f}|\mathbf{g})$ using equation (4).

From equation (1), $p(\mathbf{g}|\mathbf{f}) = p(\mathbf{w})$, the probability of having noise **w**. $p(\mathbf{g}|\mathbf{f})$ can be represented in the form

$$p(\mathbf{g}|\mathbf{f}) = \frac{1}{N} e^{-H_N(\mathbf{f},\mathbf{g})} \quad , \tag{5}$$

where $H_N(\mathbf{f},\mathbf{g})$ is the noise energy and N a normalization term. If, for example, the noise is Gaussian, $H_N(\mathbf{f},\mathbf{g}) = ||\mathbf{f}-\mathbf{g}||^2/(2\sigma^2)$, where $||\mathbf{f} - \mathbf{g}||$ is a distance measure between **f** and **g** (e.g., the number of pixels in which they differ), and σ^2 is the variance.

Since the probability of obtaining **g** alone, $p(\mathbf{g})$, is independent of **f**, it can be regarded as a constant with respect to **f**. We are thus left with the problem of determining $p(\mathbf{f})$.

Images tend to consist of large regions of constant or slowly varying brightness, separated by edges. Within a region, a pixel's brightness is expected to be similar to its nearest neighbors, so a pixel whose brightness differs greatly from its neighbors has "probably" been affected by noise (fig. 2). An image can thus be modeled as a set of pixels whose brightnesses depend on their nearest neighbors' brightness probabilistically. Such a system of interrelated probabilities is called a Markov random field [3].

It can be shown that a Markov random field is formally equivalent to an Ising [4,5] system of interacting spins on a lattice. Analogous to the Ising spin-spin interaction, a pixel-pixel interaction energy can be defined. The sum of the pixel-pixel interaction energies is the image energy, $H_0(\mathbf{f})$. Following the statistical mechanical analogy, the probability of an image **f** is determined according to the Boltzmann distribution by

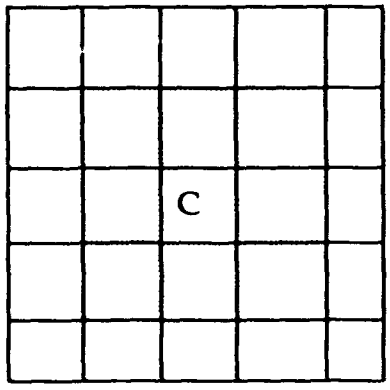

Figure 2. Markov random field model of an image. Brightness of center pixel, C, is related probabilistically to brightness levels of its nearest neighbors. Usually, 3 × 3 neighborhoods are used. In the example shown above, C is lighter than its surrounding neighbors, so it is probably "really" darker.

$$p(\mathbf{f}) = \frac{e^{-H_0(\mathbf{f})}}{Z_0} \quad , \tag{6}$$

where Z_0 is a normalization term. The pixel-pixel interaction energy thus represents our *a priori* knowledge of **f**. Finding $H_0(\mathbf{f})$, given **f**, logically and consistently is a difficult subject that will not be addressed in this paper, but it is not difficult to construct something "reasonable."

Equations (5) and (6) can now be inserted into equation (4) to yield

$$p(\mathbf{f}|\mathbf{g}) = \frac{e^{-H(\mathbf{f},\mathbf{g})}}{Z} \quad , \tag{7}$$

where $H(\mathbf{f},\mathbf{g}) = H_0(\mathbf{f}) + H_N(\mathbf{f},\mathbf{g})$ and terms that are constant with respect to **f** have been absorbed into Z. The ratio H_0/H_N indicates the importance of the *a priori* information about **f** relative to the data.

The problem of minimizing $p(\mathbf{f}|\mathbf{g})$ has thus been transformed into that of minimizing $H(\mathbf{f},\mathbf{g})$ over the space of all possible $\mathbf{f} \in S$. Since S is very large we cannot possibly examine $H(\mathbf{f},\mathbf{g})$ for all **f**. Instead we use a stochastic approach called simulated annealing.

4. Simulated Annealing

In the physical annealing process, a system is heated to a high temperature and then slowly cooled, allowing the system's atoms or spins to collectively "search" the set of all possible configurations for the one with the lowest energy consistent with the appropriate constraints. Physical annealing thus "performs" combinatorial optimization over a huge solution candidate space.

Simulated annealing mimics the essentials of the annealing process to solve the combinatorial optimization problems. Restating the problem, we seek $\mathbf{f}^* \in S$ for which $H(\mathbf{f},\mathbf{g})$ is a minimum. The essential steps of the simulated annealing procedure are the following:

1. Select a starting "temperature," T_1 (the meaning of temperature will become clear as we proceed).

2. Randomly select from S a solution candidate, $\mathbf{f}_1$; $\mathbf{f}_1$ is the current solution candidate.

3. Calculate $E_1 = H(\mathbf{f}_1)$.

4. Choose a second solution candidate, $\mathbf{f}_2$, "nearby" the current solution candidate (e.g., change one pixel of the image) and calculate $E_2 = H(\mathbf{f})_2$.

5. If $E_2 < E_1$, accept $\mathbf{f}_2$ as the current solution candidate. If $E_2 > E_1$, accept $\mathbf{f}_2$ as the current solution candidate with probability $p(\Delta E,T_1)$, where $\Delta E = E_2 - E_1$. Return to step 3, with the current solution candidate in place of $\mathbf{f}_1$.

6. After a "large" number of cycles through steps 3 to 5, decrease the temperature, and repeat steps 3 to 5 with T_1 replaced by the new temperature.

7. At "sufficiently" low temperature stop the program. The current candidate solution $\mathbf{f}'$ is the output of the process.

The acceptance probability, $p(\Delta E,T)$, is usually taken to be

$$p(\Delta E, T) = \frac{e^{-\Delta E/T}}{Z(T)} \quad , \qquad (8)$$

where $Z(T)$ is a normalization term. Temperature measures the stochastic noise that is present in the algorithm, much as it does in a physical system. At high temperatures, $\Delta E > 0$ transitions have a nonvanishing probability, allowing "jumps" over energy barriers between local minima and the global one (fig. 3). As $T \rightarrow 0$, however, $p(\Delta E,T) \rightarrow 0$, for $\Delta E > 0$, so that only $\Delta E < 0$ transitions occur and simulated annealing becomes a gradient descent algorithm.

If cooling is too rapid (quenching) the system may "freeze" in a local instead of a global energy minimum. It has been shown that $\mathbf{f}'$, the simulated annealing solution at the n^{th} iteration, converges to the global optimum, $\mathbf{f}^*$, as $n \rightarrow \infty$ for a logarithmic cooling schedule in which the temperature, T_n, at the n^{th} iteration is given by $T_n = T_0/\log(1+n)$, where T_0 is a constant. The closer T_n is to zero, the "better" the solution. Unfortunately the temperature drops very slowly (fig. 4), and there is no general error bound on the simulated annealing solution after a given finite number of iterations. There have been attempts to speed up the algorithm by using acceptance probabilities other than the Boltzmann distribution [6].

Simulated annealing can be speeded up by parallel execution on connection machines or specialized neural networks called Boltzmann machines. Further details can be found in references [7]. Alternative optimization procedures, such as mean field annealing [8], have also been proposed.

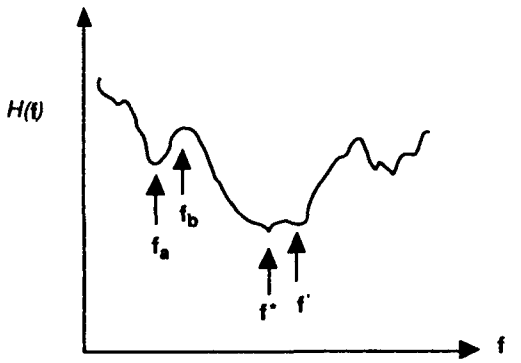

Figure 3. Goal is to determine global minimum $\mathbf{f}^*$. Gradient descent algorithms "stick" in local minima such as $\mathbf{f}_a$, so the transition $\mathbf{f}_a \rightarrow \mathbf{f}_b$ cannot occur. Simulated annealing is a stochastic procedure that sometimes allows transitions to higher energies (such as $\mathbf{f}_a \rightarrow \mathbf{f}_b$). After many iterations the simulated annealing solution is close to the global optimum.

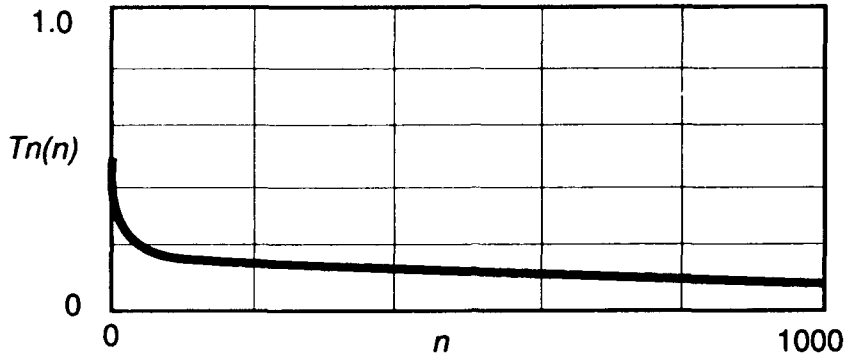

Figure 4. Normalized temperature, $Tn(n)$, after n simulated annealing iterations under logarithmic cooling schedule. $Tn(n) = T(n)/T(1)$, where $T(n)$ is the temperature at the nth iteration. $Tn(1000) = 0.1$, so temperature decreases by only a factor of 10 after 1000 cycles.

5. Image Restoration

We used 41 × 41 binary test images drawn from a set of character fonts supplied with the Turbo C programming language. The characters, together with a rectangular background, were converted into arrays of 0's and 1's. We chose the characters "A," "B," "E," "G," and "&" because they contain a variety of horizontal, vertical, diagonal, and curved line segments.

Noise was added by sequentially examining each pixel and "flipping" it (changing 1 to 0 or vice versa) with probability p, $0 < p < 1$. The decision whether or not to flip was made by drawing a random number, x, from the continuous range $0 < x < 1$ for each pixel and flipping if $x < p$. We quote the number p as the noise level in our results. Only for large images does it represent the fraction of pixels actually flipped.

The energy change, $\Delta E = H(\mathbf{f}',\mathbf{g}) - H(\mathbf{f},\mathbf{g})$, where $\mathbf{f}$ and $\mathbf{f}'$ differ by a single pixel was defined according to the number of nearest neighbors with the same brightness in a 3 × 3 neighborhood on a square grid (table 1). Edge pixels were regarded as neighbors of pixels on the opposite side of the image. We took the normalization, $Z(T)$ in equation (8) to be

$$Z(T) = \sum_i e^{-\Delta E_i} \quad , \qquad (9)$$

where the sum is over all possible values of ΔE.

Table 1. Energy change (ΔE) of central pixel (see fig. 1) flipped as a function of number of similar (same brightness) neighbor pixels (N_s)

N_s	ΔE
8,7,6	+2
5	+1
4	0
3	−1
2,1,0	−2

We performed simulated annealing by sequentially examining each pixel and deciding whether or not to flip it according to step 5 in section 4. For $\Delta E > 0$, this step was implemented by choosing a random number, x, in the range $0 < x < 1$. The pixel flip was accepted in the case $p(\Delta E,T) < x$. It might have been closer to the "spirit" of simulated annealing to examine pixels randomly, but since we added noise anew each time we ran the simulated annealing algorithm on the same image, not much was lost.

Instead of a logarithmic cooling schedule, we derived one (table 2) by inspecting the behavior of $p(\Delta E,T)$ with temperature (fig. 5). If the starting temperature is too high the simulated annealing procedure may add more noise than it removes, whereas if it is too low we are left with a simple gradient descent algorithm. We used a total of only 10 annealing cycles.

Table 2. Simulated annealing temperature schedule

Temperature	No. of cycles
100.0	1
50.0	1
20.0	1
5.0	1
1.2	5
0.6	1
0.3	1

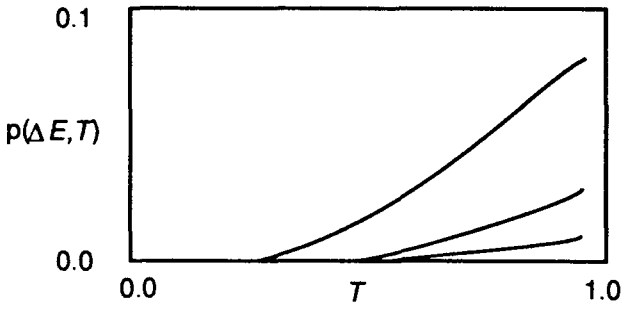

Figure 5. Behavior of transition acceptance probability, $p(\Delta E,T)$, with temperature for different values of ΔE. As $T \rightarrow 0$, $p(\Delta E,T) \rightarrow 0$; thus at low temperature, simulated annealing becomes a gradient descent.

Restoration quality was studied by running the program repeatedly on the sets of noisy images with "fresh" noise inserted each time. Restoration quality was defined as the fraction of pixels that were different between the restored and the original image. Our procedure was the following: (1) choose a character and corrupt it with noise level p, (2) apply the simulated annealing algorithm, and (3) measure the quality of the restoration. The procedure was run 100 times for each character and statistics were compiled.

6. Results and Discussion

Our results are summarized in table 3, and in figure 6, where we quote the mean and standard deviation of the restoration quality for 100 runs on each character. An example of an original image corrupted with noise and then recovered is shown in figure 7.

For noise levels less than about 10 percent, our simulated annealing procedure does not improve image quality, and in fact may actually degrade it. This is not surprising because annealing is a stochastic process. Ideally the temperature schedule should be tailored to the noise level in each original image, but we used one schedule for all images. Many more efficient techniques exist for extracting patterns out of low noise backgrounds, so simulated annealing would be of little use in this regime anyway.

As the data show, our simulated annealing procedure did yield significant improvements in image quality in the 20- to 30-percent noise range. While the behavior of the algorithm differed from character to character, the differences were not large.

These results are quite promising in view of the fact that we used only 11 annealing cycles with an *ad hoc* temperature schedule. For future studies we plan to implement a logarithmic cooling schedule and run more cycles. We also plan to examine the effect of different acceptance probability distributions, $p(\Delta E,T)$, such as the one proposed by Szu and Hartley [6].

Table 3. Average restoration quality and standard deviation for 100 simulated annealing trials per character (procedure of sect. 3.3)

Initial distortion	0%	10%	20%	30%
Character				
A	9 ± 1%	9 ± 1%	10 ± 1%	14 ± 2%
B	10 ± 1%	10 ±1%	11 ± 1%	15 ±2%
G	10 ± 1%	10 ±1%	11 ± 1%	15 ±2%
E	10 ± 1%	10 ± 1%	11 ± 1%	14 ± 2%
&	11 ±1%	11 ±1%	12 ± 1%	16 ±2%

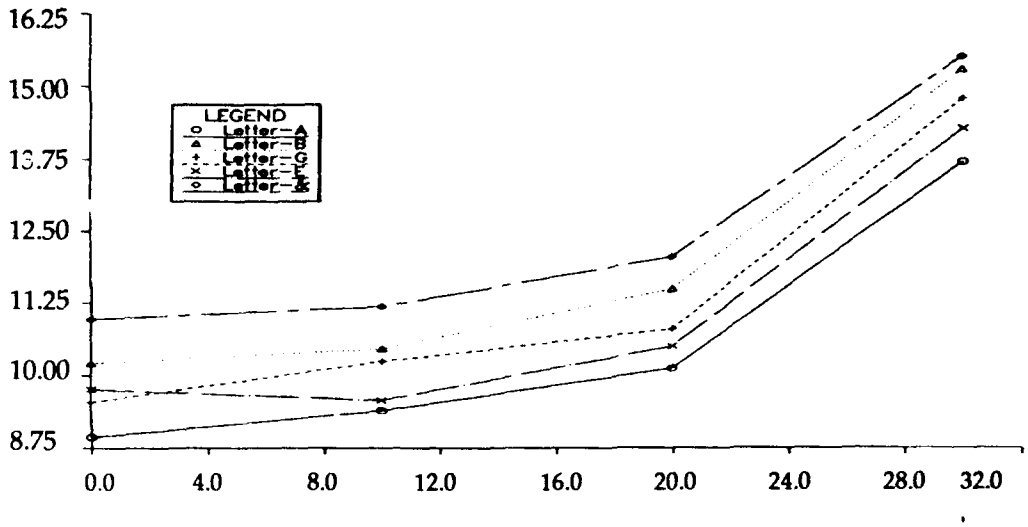

Figure 6. Noise remaining after simulated annealing as a function of initial noise level for each symbol.

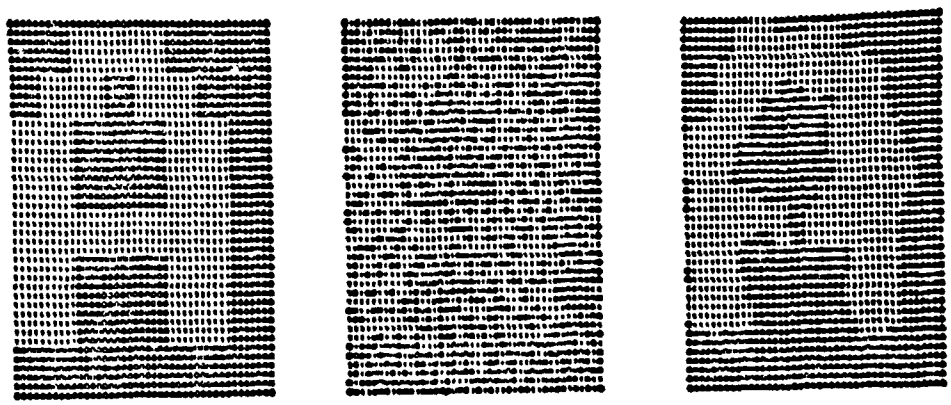

Figure 7. Image corrupted with noise and recovered: (a) original image, (b) with 30-percent noise, and (c) after 10 simulated annealing runs 11 percent of the pixels differ from the original.

References

1. K. Sam Shanmugan and A. M. Breipohl, *Random Signals: Detection, Estimation and Data Analysis*, John Wiley and Sons (1988).

2. M. R. Garey and D. S. Johnson, *Computers and Intractability*, W. H. Freeman (1979).

3. S. Geman and C. Graffigne, *Proceedings of the 1986 International Congress of Mathematics*, A. M. Gleason, ed., Am. Math. Soc. (1987).

4. M. Toda, R. Kubo, and N. Saito, *Statistical Physics I*, Springer Verlag (1986).

5. S. Geman and D. Geman, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, **PAMI-6** (November 1984), 721.

6. H. Szu and R. Hartley, *Phys. Lett.*, **A 122** (1987), 157.

7. S. Kirkpatrick et al., *Sci.*, **220** (13 May 1983), 671.

P. van Laarhoven and E. Aarts, *Simulated Annealing: Theory and Applications*, D. Reidel (1987).

E. Aarts and J. Korst, *Simulated Annealing and Boltzmann Machines*, John Wiley and Sons (1989).

8. H. Hiriyannaiah et al., *J. Optical Soc. of America*, **A6** (1901).

DISTRIBUTION

Administrator
Defense Technical Information Center
Attn DTIC-DDA (2 copies)
Cameron Station, Building 5
Alexandria, VA 22304-6145

University of Maryland
Computer Science Department
Attn Dr. S. Lomanaco
Attn Dr A. Sherman
Attn Dr. C. Nicholas
Attn Dr. James Lyle
Baltimore County
Catonsville, MD 21228

Night Vision & Electro-Optics Center
Attn D. Singer, AMSLC-RD-NV-ISP
FT. Belvoir, VA 22060

U.S. Army Missile Command
Attn J. Johnson, AMSMI-RD-WS-PO
Redstone Arsenal, AL 35809

Dr. J. Cole
A-Higashi, Sekishin Mune
2-12-19 Sekinach-Minami
Nerima-Ku
Tokyo 177
Japan

IIT Research Institute
Ohio Technology Center
Attn S. Worrell
4140 Linden Avenue, Suite 201
Dayton, OH 45432-3018

US Army Laboratory Command
Attn Technical Director, AMSLC-TD

Installation Support Activity
Attn Legal Office, SLCIS-CC

USAISC
Attn Record Copy, AMSLC-IM-VA
Attn Technical Reports Branch,
AMSLC-IM-VP (2 copies)

Harry Diamond Laboratories
Attn D/Laboratory Directors
Attn Library, SLCHD-TL (3 copies)
Attn Library, SLCHD-TL (Woodbridge)
Attn N. Berg, SLCHD-S3
Attn A. Sindoris, SLCHD-S3
Attn B. Weber, SLCHD-S3
Attn J. Goff, SLCHD-ST-OP
Attn R. Johnson, SLCHD-ST-OP
Attn J. Mait, SLCHD-ST-OP
Attn D. McGuire, SLCHD-ST-OP
Attn J. Pellegrino, SLCHD-ST-OP
Attn M. Younger, SLCHD-ST-SS
Attn M. Patterson, SLCHD-ST-SS
Attn D. Gerstman, SLCHD-ST-OP (40 copies)